قبیله گیک ها



سال دوم ، شماره بیست و چهارم ، آبان ۹۶

تنها مجله مخصوص گیک های ایرانی



قبیله گیک ها دو ساله شد





Geeks Tribe

قبیله گیک‌ها

تنها مجله مخصوص گیک‌های ایرانی

سال دوم - شماره ۲۴ آبان ماه ۱۳۹۶

در این شماره میخوانیم:

نویسندگان این شماره

بابز

بردیا

شیرین ابراهیمی

رهام مصلی

کیا حامدی

حسین شفیعیان

سیروس فتح اللهی

محمد رضا لامعی

جعفر آخوندعلی

امین سامانی

Www.Geekstribemedia.com

info@geekstribemedia.com

سخن سردبیر



با درود و آرزوی موفقیت برای شما همراهان همیشگی قبیله گیک‌ها با شماره ۲۴ در خدمت شما هستیم و امیدواریم بتوانیم بار دیگر ساعتی شما را با خود همراه کنیم.
آبان ماه امسال قبیله گیک‌ها وارد سوم سال فعالیت خود شد و خوشحالیم که در این مدت موفق شده‌ایم جامعه‌ای بزرگ از گیک‌های فارسی زبان را به دور هم جمع کنیم تا در پلتفرمهای مختلف فضای‌مجازی اعم از(تلگرام،توئیتر،فیسبوک،یوتیوب،آپارات،اینستاگرام ، دیسکورد و...) با یکدیگر به تبادل اطلاعات بپردازند و از تجربیات یکدیگر استفاده کنند.
در این دو سالی که از تشکیل قبیله گیک‌ها میگذرد دوستان زیادی به ما در رسیدن به هدفمان که همانا برپایی جامعه‌ای پویا و مولد اطلاعات بود کمک‌های فراوانی کردند. بعضی از این دوستان بنا به دلایل مختلفی بعد از مدتی از جمع ما رفتند و دوستان جدیدی جایگزین آن‌ها شدند و بعضی دیگر از دوستان همچنان در کنارمان هستند و مشتاقانه و داوطلبانه به هر نحوی که بتوانند به قبیله گیک‌ها و اهدافش کمک میکنند.
دوستان زیادی در این مدت برای قبیله گیک‌ها در اقدام به تولید محتوای (نوشتاری،صوتی،تصویری)گیکی کرده‌اند بعضی از این مطالب در قالب یک مطلب منتشر شده‌اند و بعضی دیگر بصورت سلسله مطالبی منتشر شده‌اند و همچنان نیز ادامه دارند.
قبیله گیک‌ها همیشه بر این موضوع تأکید داشته است که سعی میکند فقط مطالبی را منتشر نماید که تولید شده توسط نویسندگان آنِ باشد که متأسفانه در گذشته بدلایل مختلف در بعضی موارد مطالبی در قبیله گیک‌ها منتشر شده است که قبلاً در سایتهای دیگر منتشر شده بوده‌اند.
بهمین دلیل تصمیم گرفته‌ایم در سال سوم فعالیتمانٍ فقط مطالبی را منتشر نمائیم که برای اولین بار در فضای مجازی از طریق قبیله گیک‌ها منتشر شده‌اند. احتمالاً به همین دلیل در چند ماه اول سال سوم شامل تقلیل تعداد مطالب در هر شماره خواهیم بود که امیدواریم با همکاری شما دوستان عزیز بتوانیم تعداد مطالب را بیشتر کنیم.
یکی از اخبار خوبی که در این ماه برای شما دوستان عزیز داریم بازگشت دوباره دوست خوب و فعالمان آقای محمد مهدی خلعتبری مدیر بخش ویدیوکست قبیله گیک‌ها به جمعمان است که امیدواریم با حضور فعال و همیشگی خود دوباره شاهد فعالیت بیشتر بخش ویدیوکست قبیله گیک‌ها در آینده باشیم.

در پایان و مثل همیشه باز هم از همه شما دوستان و همراهان همیشگی صمیمانه درخواست میکنیم تا با ارسال نظرات، پیشنهادات و انتقادات خود ما را در هرچه بهتر شدن قبیله گیک‌ها یاری نمائید.

نویسنده : بابز

صفر و یک



01110100 01101001 01110010 01100011 00100000 01100101 01101000 01010100
01101001 01101000 01110100 00100000 01101100 01100001 01100011 01101001
01100101 01100100 00100000 01101110 01101001 00100000 01100111 01101110
01100111 01101110 01101001 01110000 01101111 01101100 01100101 01110110
01110010 01100001 01110111 01110100 01100110 01101111 01110011 00100000
01110100 01101111 01101110 00100000 01110011 01101001 00100000 01100101
01101111 01110010 01110000 00100000 01100101 01101000 01110100 00100000
01110100 01101001 00100000 00101100 01101101 01100001 01110010 01100111
01100100 00100000 01100101 01101000 01110100 00100000 01110011 00100111
01001001 00100000 00101110 01101110 01100111 01101001 01110011 01100101
01100001 01110010 01110100 00100000 01110011 01101001 00100000 01110100
01100111 01101110 01101001 01110100 01100001 01101100 01110011 01101110
01110100 01110011 01110010 01100101 01100100 01101110 01110101 00100000
01101111 00100000 01100111 01101110 01101001 01100100 01101110 01100001
01101110 00100000 01110010 01100101 01110011 01110101 00100000 01100110
01110100 01101110 01101001 00100000 01110011 01100100 01100101 01100101
01101000 01110100 01100101 01101101 01101111 01110011 00100000 01101111
01110100 01100001 01101000 01110100 00100000 01100111 01101110 01101001
00100000 01100101 01100010 00100000 01101110 01100001 01100011 00100000
01100100 01100101 01111010 01101001 01101100 01100001 01100101 01110010
01101111 01100011 00100000 01100001 00100000 01110011 01100001 00100000
01110000 00100000 01110010 01100101 01110100 01110101 01110000 01101101
00101110 01101101 01100001 01110010 01100111 01101111 01110010

تهیه کننده: بابز

تولد دو سالگی قبیله گیک ها

هر پاییز نوید آغاز ماست و چه خوش بود این دومین پاییز



طی دو سال گذشته، قبیله گیک ها از مجله‌ای مختص مطالبِ کامپیوتری تبدیل به مجموعه‌ای شامل مقالات، بخش ویدیوکست های آموزشیِ کامپیوتری و بخش پادکست های موسیقی شد، سایتِ این مجموعه توسط دوستان خوبمون طراحی و مدیریت شد و بعد از آن، بخش فروشگاه قبیله گیک ها راه اندازی شد و تلاش کردیم مجموعه‌ای در حد توان و تلاشمون برای شما عزیزان فراهم کنیم تا بتونیم رضایت شما و دل خودمون رو به دست بیاریم.
همه ما برای رسیدن به اهدافمون با هم و در کنار هم، برای هرچه پربار شدن این مجموعه تلاش کردیم و از حمایت ها و همچنین پیشنهاد ها و انتقادهایِ شما عزیزان نهایت بهره رو می بریم. در ادامه به شرح مراسم تولد دو سالگی قبیله گیک ها می پردازیم، از همراهی شما بزرگوارانِ سپاسگزارم:

اولین روز شروع همکاری با قبیله گیک ها رو هیچ‌وقت فراموش نمیکنم، روزی که بابز برای نویسندگی در مجله از من دعوت کرد و من با استرس اینکه تا به حال برای هیچ مطلب علمی ای دست به قلم نشدم، (از وقتی به خاطر میارم نویسندگی رو دوست داشتم و همیشه همه محتویات ذهنم رو یادداشت میکردم، شاید بهتره اینطور بگم که لوازم اتاق من در هر گروه سنی حتماً دو یا سه تا دفترچه یادداشت خوشگل داشت که برای من همه چیز بودن) دو دل بودم که چه جوابی باید بدم، بی درنگ محتویات ذهنم رو در پاسخ گفتم و بابز با شکیبایی هرچه تموم تر این جمله رو جواب داد: "بیا ما کمکت میکنیم" و از همون روز برای نوشتن اولین موضوع دست به کار شدم و تلاش کردم همه چیز خوب پیش بره.
همون روزهای اول موضوعی که توجه من رو به خودش جلب کرد محیط کاملاً دوستانه قبیله بود، ما بارها و بارها تو جلسات مختلف تصمیم می گرفتیم تا روش جدیدی رو اعمال کنیم اما بعضی وقت ها جوری که می خواستیم پیش نمیرفت و دوباره و دوباره تست می کردیم. توی این راه حسابی خسته شدیم و هنوز هم خیلی موارد دیگه ای هست که باید اجرایی بشه، خیلی‌ها شاید نتونستن دووم بیارن و متأسفانه با توجه به مخالفت جمع از ما جدا شدن، خیلی‌ها خواستن برن و ما زورمون رسید و اجازه ندادیم و اما خیلی از دوستان به این جمع اضافه شدن و ما از حضورشون دلگرم شدیم.

تو اولین شماره حضورِ من، خیلی همه چیز سخت گذشت، بدون اینکه یک بار هم تجربه ویرایش متن مجله رو داشته باشم برای عرق به قبیله قرار شد تو این موقعییت بحرانی مسولیت کار رو به عهده بگیرم و دقیقا یادمه که فقط سه روز فرصت داشتم و یه مشکل اَساسی تو اون سه روز، تمپلیت سه ستونه (اونو یادتونه حتماً :)) مجله بود که هیچ جوری نمیشد درست سرهمش کرد و در نهایت شد، اما بعد ها متوجه شدم متن یکی از هم گروهی هامون نصفش تو مجله بود و نصف دیگه اش متاسفانه نبود و اما به دلیل اون حجم کار دوستان لطف کردن و هیچ چیزی رو به روی من نیاوردن:)
از اون روز دوسال گذشت و برای من قبیله همیشه شبیه به کودکیه که داره جلوی چشمای خودم پا میگیره و بزرگ میشه و من هر لحظه میبینم چطور نیازهاشو از ما طلب میکنه و هیچوقت این حس تموم نمیشه و هرسال برای تولدش هر کاری بتونم انجام میدم چون وجودش برای من با ارزشه.
شاید برای شمایی که این متن رو می خونید این حس خیلی با معنی نباشه، اما باید باشید و خودتون ببینید، که چطور چند صفحه مجازی به وجود میارید و بعد مجله شدن، سایت شدن، مجموعه شدن، بزرگ شدن، بزرگ شدن و بزرگ شدنش رو می‌بینید و چه لذتی از این بالاتر که دسترنج تلاش و همدلی خودت و دوستات اینجور وِرد زبونِ خیلی‌ها میشه و تو حتماً لذت می‌بری از این پیشرفت و نتیجه مثبت.

خب بریم سراغ تولد، برنامه ریزی ها از اوایل تابستون شروع شد، خیلی مهم بود که همه چیز خوب پیش بره و قبیلهِ عزیز ما باز هم باعث افتخار خودش و ما بشه، به همین خاطر یکی از بهترین رستوران های بام لند در محدوده دریاچه چیتگر با کیفیت عالی و فضایی بی نظیر انتخاب شد تا برای صرف ناهار به اونجا بریم و بعد از اون به کافی شاپی زیبا برای صرف کیک و چای رفتیم.
هدف از برگزاری این جشن، شادی یک سال تلاشِ پر ثمر، با هم بودن برای بزرگ شدن قبیله و آشنایی با دوستان جدید بود. دوستان همیشگیمون با گرمای حضور پر مهرشون جشن کوچکمون رو صفایی تازه بخشیدن.
بهتره همین جا، جای تک تک دوستانی که نتونستند تشریف بیارن رو خالی کنم و پیشنهاد کنم که برای سال بعد حتما در کنار ما این روز خاص رو جشن بگیرن و شادی های این اتفاق خوش رو دو چندان کنن.
بگذریم از اینکه چقدر خوش گذشت و دیداری تازه شد و قراری بود پر از حرف‌های نگفته و بحث‌های ملسِ کار :)

امسال تصمیم بر این شد که به دلیل غیبت های گاه و بی گاه مدیر بخش ویدیوکست ها جناب آقای خلعتبری و به پاس قدردانی از زحمت های بی دریغ ایشون، افتخار برش کیک نصیب ایشون بشه و این دومین جشن پاییزی رو به زیبایی و به یاد تک تک شما دوستان عزیز به پایان رسوندیم.



نویسنده : شیرین ابراهیمی

آرچ را یکبار تجربه کنیم

آرچ لینوکس توزیع دوست‌داشتنی و جذاب درعین حال خشن و بی اعصاب



میخواهیم در این مقاله بگوییم چرا باید یکبار آرچ را امتحان کنیم.
واقعیت اینه در دنیای گنو/لینوکس افراد یا عاشق آرچ هستند و هیچ توزیع دیگری را قبول ندارند یا روی توزیع های دیگه هستند و آرچ را توزیع بیخود میدانند که باعث هدر دادن وقت است.
اما واقعا چرا بعضی ها عاشق آرچ لینوکس هستند ولی بعضی ها اصلا ازش خوششون نمیاد؟
بهتره قبل از اینکه به این سوال جواب بدیم یکم بیشتر با آرچ آشنا بشیم
آرچ لینوکس اسم یک توزیع گنولینوکسیه که هدف اصلیش سادگی و سبک بودنه
اما اینا یک سری کلمات کلیشه ای شده سادگی و سبک بودن، باید بفهمیم چطوری؟ چطور توزیع آرچ تونسته سادگی و سبک بودن رو ملاک اصلیش قرار بده؟
خب جوابش سادست با نداشتن هیچی
شاید براتون عجیب باشه چطور هیچی؟ خب آرچ درواقع هیچی نداره حتی یک بسته ساده چند خطی، درواقع شما در آرچ از هیچی یک سیستم عامل میسازید با کنار هم قرار دادن تک تک بسته ها، وای جالب شد و البته کمی ترسناک
منظور از هیچی این نیست که هیچ بسته و پکیجی برای این
سیستم عامل نوشته نشده بلکه منظور اینه مثل یک ساختمون که اول هیچی وجود نداره و کم کم با ساختن قسمت های مختلف ساختمان شما شکل میگیره در آرچ هم خود کاربرا باید بسته های مورد نیاز خودشون رو کنار همدیگه قرار بدند و سیستم عامل خودشون رو شکل بدند.

شاید این بخش برای شما کمی گنگ باشه بزارید براتون بازترش کنم
سیستم عاملی مثل اوبونتو یا مینت رو در نظر بگیرید که شما فایل رو از سایتش دانلود و اقدام به نصب میکنید چندتا سوال ازتون میپرسه مسیر نصب و در انتها بعد از ساختن نام کاربری و کلمه عبور سیستم عامل شما نصب میشه یعنی در واقع خود شرکت بسته های مورد نیاز برای اجرای معمولی سیستم عامل رو در کنار همدیگه قرار داده و شما فقط بهش یک مسیر میدید که در اونجا کپی بشه حتی اگه کامپیوتر شما بلوتوث نداشته باشه بسته های راه اندازی بلوتوث روی سیستم عامل شما وجود داره و کلی بسته دیگه که ممکنه هیچوقت ازشون استفاده نشه و درواقع بسته های اضافی روی سیستم عامل شما هستند که باعث کند شدن سیستم عامل میشند البته شاید بگید با مشخصات بالای سخت افزار های امروزی این کندی به چشم نمیاد اما خب باز هم بسته های اضافی روی سیستم عامل شما وجود داره که ازشون استفاده نمیشه
خب کاری که آرچ کرده دادن حق انتخاب به کاربره تا کاربرانش بتونند بسته های مورد نیاز خودشون رو در کنار هم دیگه قرار بدند و سیستم عاملشون رو شکل بدند البته باید بگم در این بخش هست که بعضی افراد عاشق آرچ میشند و بعضی ها متنفر، کسایی که بتونند کامل بسته هارو در کنار همدیگه قرار بدند به نوعی شیفته آرچ میشند و نصب سیستم عامل های دیگه برای اونها لذتی نخواهد داشت چون سوار شدن ماشینی که یکی دیگه درست کرده باشه با سوار شدن ماشینی که خودتون درست کرده باشید خیلی باهم فرق داره و بخش دوم افرادی که موفق به نصب آرچ نشده و با گیجی و دلسردی برمیگردند سراغ یکی از توزیع های دیگه
دادن این قابلیت به سیستم عامل باعث شده آرچ لینوکس یک توزیع ساده و سبک باشه
سادگی و سبکی به اندازه نداشتن یک رابط گرافیکی تا اندازه ای که مدیر بسته آرچ هم بدون رابط گرافیکی هست البته این یکی از خوبی های آرچه که کاربرا با سلیقه خودشون رابط های گرافیکی رو نصب میکنند
خب تا اینجا فهمیدیم که آرچ سیستم عاملی هست که در ابتدا هیچ چیزی نداره و در انتها بسته های روی اون بسته های مورد نیازه کاربره.
حالا باید بفهمیم آرچ جدا از سادگی و سبکی چه خوبی برای کاربرانش داره چون با سخت افزار های الان کند بودند سیستم عامل ها خیلی به چشم نمیاد (البته جدا از ویندوز)
خوبی که آرچ داره یاد دادن بخش های زیری سیستم عامل به کاربره، کسی که آرچ رو نصب میکنه با خیلی بخش های سیستم عامل آشنا میشه، البته دانستن این اطلاعات در مورد بخش های مختلف سیستم عامل برای افراد معمول که هدفشون فقط استفاده از سیستم عامله مهم نیست چون در صورت خرابی هر سیستم عاملی دوباره اون رو از اول نصب میکنند یا به کسی که بلده مراجعه میکنند



ولی برای افرادی که علاقه دارند از ریز به ریز سیستم عاملشون آگاهی داشته باشند آرچ لینوکس یک رویای قابل لمسه
در اینجا ما به کسی نمیگیم آرچ لینوکس رو انتخاب کنه چون آرچ لینوکس نیازمند وقت خیلی زیادیه تا بخش بخش اون کنار همدیگه چیده بشه و صد البته وقت بیشتر که مشکلات اون برطرف بشه، خیلی افرادی که به سراغ آرچ میان با هدف یادگیری لینوکس و بخش های مختلف اون میان تا بخشی از سیستم عامل دچار ایراد بشه و آنها مشکل رو بطرف کنند که به واسطه این کار اطلاعاتی در مورد اون بخش کسب میکنند، این کار نیازمند وقت زیادیه پس آرچ رو نباید به افرادی که مشغله کاری زیادی دارند
پیشنهاد کرد برای کسی که برنامه نویسه وقتش با ارزش تره که برای برنامه نویسی گذاشته باشه تا برای حل مشکلات آرچ.

بیاید در مورد یک امکان خوب آرچ صحبت کنیم، قابلیت رولینگ ریلیز بودن آرچ

رولینگ ریلیز بودن به چه معناست و چه تفاوتی میان سیستم عامل هایی که رولینگ ریلیز هستند با دیگر سیستم عامل ها وجود داره؟
برای جواب این سوال شما سیستم عامل ویندوز رو در نظر بگیرید که هر چند سال یک سیستم عامل با اسم جدید و امکانات و آپدیت های جدید به بیرون عرضه میشه برای مثال ویندوز ۷ و بعد از اون ۸ و ۱۰ اومدند و باید منتظر نسخه های بعدی با اسم های مختلف و ظاهر های متفاوت هم باشیم، تا اینجای کار رو همه بلد هستیم که هر نسخه از ویندوز بیاد ما از اطلاعاتمون یک پشتیبان تهیه میکنیم و یا اطلاعات رو به پارتیشن های دیگه انتقال میدیم و سیستم عامل قدیم رو پاک و نسخه جدید ویندوز رو نصب میکنیم و دوباره شروع به نصب درایور ها و برنامه های خودمون روی اون میکنیم. این روش در سیستم عامل های ویندوز و بسیار از توزیع های گنو/لینوکسی یافت میشه

که بعد از رسیدن به یک بهینگی نسخه جدید رو منتشر میکنند اما در آرچ شما نسخه ای از اون رو ندارید شما کافیه یک بار آرچ رو نصب کنید تا از نصب مجدد سیستم عامل و نسخه های جدید خلاص بشید این قدرت رولینگ ریلیز بودنه که شما یک بار نصب میکنید و برای همیشه استفاده میکنید ،هربار کافیه فقط بسته های نصب شده روی سیستم عامل رو به آخرین نسخه های موجود آپدیت کنید شما لازم نیست منتظر انتشار نسخه جدید آرچ باشید تا قابلیت های جدید یا آپدیت های مختلف رو داشته باشید یک بار آرچ را نصب کنید همه قابلیت هارو خواهید داشت درواقع انتشار آرچ لینوکس یک تصویر لحظه ای از از بروزرسانی بسته های موجود با بسته های کنونی است تا شما نسخه نهایی آرچ را داشته باشید.



خب با دانستن این اطلاعات از آرچ حالا باید به این سوال جواب بدیم که آرچ لینوکس برای چه افرادی مناسبه؟
در جوابش باید گفت ما آرچ رو به مهندسین معمارها گرافیستان برنامه نویسان و افرادی که کارشون از سیستم عامل استفاده از یک یا چند نرم افزار خاصه و جز اون کاری دیگه ای با سیستم عامل ندارند پیشنهاد نمیکنیم، درواقع آرچ رو به افرادی که وقت برای کل انداختن با سیستم عامل دارند و علاقمندند به زندگی کردند در دنیای تاریک ترمینال برای حل مشکلات آرچ، پیشنهاد میکنیم این افراد باید یک بار آرچ را امتحان کنند.

نویسنده : کیا حامدی دانشجوی ارشد نرم افزار، علاقمند به برنامه نویسی و دنیای نرم افزارهای آزاد

Ansible

درنهایت یه زمانی میرسه که شما سرور یا سرور های خودتون رو کانفیگ کنید : سرورهایی را که نیاز دارید اون ها رو کانفیگ کنید در شرایط عادی باید به هر کدوم SSH بزنید و دستی کانفیگش کنید ممکنه ۱۰۰ دستور و هرکدوم و دستی توی ترمینال وارد کنید مثال : نصب کردن بسته ها، ویرایش فایل های پیکربندی و غیره، اما این داستان میتونه خیلی زمان شما رو بگیره و اعصاب شما رو خورد کنه ممکنه بعد از مدتی کارایی شما کم شه ، اینجاست که خطاهای انسانی وارد میشه و کل داستان و خراب میکنه !!! یا حتی میتونه برای ادمین خسته کننده باشه که یه سری کار های روتین و تکراری و روی چندین سرور انجام بده، تصور کنید یک سری Task های پیچیده دارید مثل داشتن یک OpenStack cloud در برنامه خودتون ، انجام این کارها به صورت دستی دیوانگی بیش نیست .اگر شما در یک شرکت درست و حسابی کار کنید ،مدیر ارشد سیستم اگر متوجه شه شما دارید کل دستورها رو دستی خودتون وارد میکنید ممکنه عصبانی بشه و شمارو اخراج کنه ، بله این کاملا درسته در بیزینس های واقعی کسی با شما شوخی نداره.
حتما متوجه این شدید که برای حل این مشکل شما نیاز به ابزار های مثل configuration management دارید یه چیزی مثل Asnible یا Pupet ، این ابزار های configuration management tool میتونه راه حل مناسبی برای شما باشه، اگر شما Developer هستین یا SysAdmin و دنبال یه ابزار خوب Automate میگردین به نظر من بهترین ابزار برای شما Ansible خواهد بود. اینکه فرقش با بقیه ابزار ها چیه بمونه برا بعد، اما Ansible چیه ؟

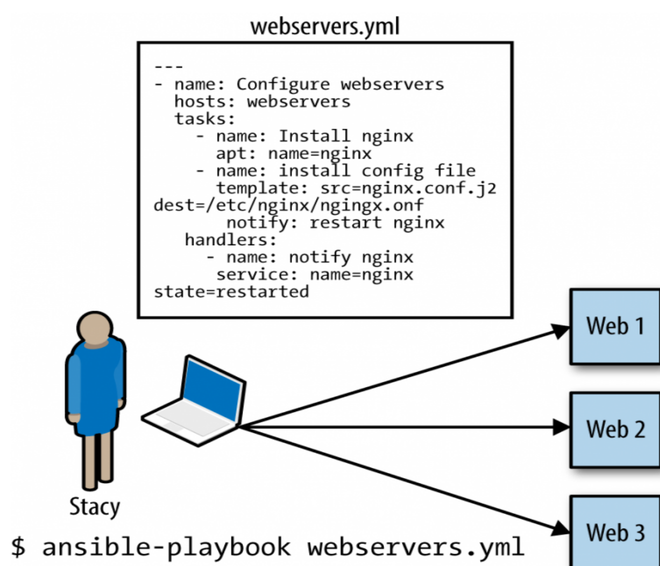
Ansibleچیست

این اسم یه اسم علمی تخیلیه گرفته شده از یک کتاب هست. **ansible** یک دستگاه ارتباطی تخیلی است که می تواند اطلاعات را سریعتر از سرعت نور انتقال بده که نویسنده Ursula K. Le Guin این مفهوم را در کتابش به اسم Rocannon's World اختراع کرد، و نویسندگان دیگه داستان های علمی تخیلی این ایده رو از Le Guin گرفتند.
دقیقتر بخوام بگم ، Michael DeHaan خالق Ansible این اسم رو از نام کتاب "Ender's Game" که توسط Orson Scott Card نوشته شده گرفته (در سال ۲۰۱۳ نیز فیلمی با این نام توسط Gavin Hood ساخته شد)
در" کتاب Ender's Game" اسم ansible برای کنترل تعداد زیادی از کشتی های راه دور در یک زمان، در فواصل خیلی وسیع استفاده میشده.
حال به عنوان یک استعاره از این اسم برای کنترل سرورها از راه دور استفاده میشود.



ابزار Ansible چطور کار میکنه ؟

تصویر زیر نمونه مورد استفاده Ansible را در عمل نشان می دهد.



یه یوزر به اسم Stacy از Ansible برای کانفیگ کردن ۳ سرور مبتنی بر Ubuntu استفاده میکنه تا بتونه Nginx رو روی اونها اجرا کنه. Stacy یک اسکریپت Ansible نوشته به اسم webservers.yml ، در Ansible به یک اسکریپت playbook گفته میشود.یک playbook تعریف میکنه که کدوم هاست (دوراقع در Ansible به اسم remote servers شناخته میشه) باید برای کانفیگ استفاده بشه. و یک فهرست مرتب شده از وظایف برای انجام در آن سرورها را شامل میشود. در این مثال هاست های ما web۱، web۲ و web۳ هستند و کارای که در آنها باید انجام بشه اینا هستند :

۱- نصب nginx
۲- ایجاد فایل کانفیگ nginx
۳- استارت کردن سرویس nginx

در مقالات بعدی صحبت خواهیم کرد که Stacy چطور این playbook را اجرا میکنه ، فعلا در این مثال playbook با نام webservers.yml ذخیره شده که میشه با دستور زیر اون رو اجرا کرد

```
$ansible-playbook webservers.yml
```

Ansible درواقع یک ارتباطات SSH را به صورت موازی با web۱, web۲, and web۳ ایجاد میکند در نهایت اولین دستور در لیست هر سه سرور را به صورت همزمان اجرا خواهد کرد در این مثال اولین دستور نصب پکیج nginx توسط دستور apt (درواقع) میباشد Ubuntu بنابراین دستور (از این پکیج منیجر برای نصب بسته ها استفاده میکند playbook برای این کار چیزی مانند دستور زیر است.

```
-name: install nginx
Apt: name=nginx
```

همچنین Ansible بعد از اجرای یه سری کارای دیگه هم میکنه:

۱. یک اسکریپت پایتون ایجاد میکنه که بسته nginx را نصب می کنه.
۲. اسکریپت را به web۱، web۲ و web۳ کپی میکند.
۳. اسکریپت را در web۱، web۲، web۳ اجرا میکند.
۴. صبر میکنه تا اسکریپت برای تمام موارد اجرا شود.

سپس سراغ کار بعدی در لیست حرکت میکنه و از طریق همین چهار مرحله آنها را انجام میده. خیلی مهمه که توجه داشته باشید:

Ansible هر کار را به طور موازی در تمام میزبان اجرا می کند.
Asnible منتظر می ماند تا تمام هاست ها قبل از رفتن به کار بعدی کار قبلی را تمام کنند.
Ansible وظایف را به ترتیبی که آنها را مشخص می کنید اجرا می کند.

امیدوارم براتون مفید باشه، هر سوالی داشتن میتونید در گروه قبیله در تلگرام مطرح کنید

نویسنده : رهام مصلی

آزادی در سیستم دانشگاهی سما!

سیستم دانشگاهی سما چیه؟
سیستم سما یک سیستم دانشگاهی برای مدیریت انتخاب واحد، ارزشیابی استادان و در کل انجام کار های آموزشی هست و اکثر دانشگاه‌های کشور دارن از این سیستم استفاده میکنن!

مشکل این سیستم چیه؟
یکی از مشکلاتی که این سیستم برای ما دانشجو ها بوجود میاره ارزشیابی اساتیده! قبل از اینکه ترم تموم بشه و بخوایم امتحان بدیم تا یه زمانی وقت داریم که ارزشیابی اساتید رو انجام بدیم. در ارزشیابی برای هر استاد حدود ۳۰ تا سؤال وجود داره، برای رشته‌های مهندسی خیلی زیاد نیست ولی برای رشته های پزشکی خیلی زیاد بود طوری که دوستم میگفت ما ۱ روز باید وقت بذاریم تا این ارزشیابی رو پر کنیم.
اگه تو اون زمان ارزشیابی نکرده باشیم، ترم بعد که میخوایم انتخاب واحد کنیم سیستم برای ما دیرتر از بقیه دانشجو ها فعال میشه و کلاسا پرمیشن و ما مجبور میشیم ترم بعد بگیریم یا تلاش کنیم اضافه بر ظرفیت بگیریم!

راه حل ؟
افزونه ای براتون آماده کردم که با استفاده از اون میتونید صرفاً با تعیین درصد رضایتتون از استاد، تمام گزینه هارو با یک کلیک علامت بزنین ! یعنی شما میگین از این استاد ۵۰٪ رضایت دارین، افزونه برای تمام سؤالات نصف گزینه هارو در نظر میگیره.
از اساتیدمون درباره پرکردن همه گزینه ها بصورت یکسان پرسیدم و اونها بهم گفتن این سیستم زمانی که همه ی گزینه ها روی حالت کمترین یا بیشترین نمره باشن ارزشیابیتون حذف میشه. برای همین یه گزینه دیگه گذاشتم برای دور زدن فیلتر آماریشون که فعال باشه میاد به صورت تصادفی یه مقداری که از امتیاز رو کم یا زیاد میکنه و اینی که حالا چند گزینه در میون این اتفاق میوفته بازم رندومه جوری که ردیابی نشه. یه گزینه دیگه هم وجود داره که اگه فعال باشه، اطلاعات فرم ارسال هم میشن، میتونین این گزینه رو غیر فعال کنین تا بتونین تغییراتی روی اطلاعات بدین.



دیگه چیکار میشه کرد؟
یکی دیگه از مشکلاتی که دانشجو ها هرترم دارن، ساخت برنامه درسیه. هرترم بعد از انتخاب واحد دانشجو ها باید یه برگه بگیرن دستشون و بنویسن هر کلاس قراره کجا برگزار بشه، چه ساعتی و ...
یه قابلیت دیگه ای که این افزونه داره تولید برنامه هفتگیه. کافیه دانشجو بره داخل صفحه واحدایی که انتخاب کرده، و بعد توی افزونه گزینه ساخت برنامه درسی رو بزنه تا برنامه درسی خودش رو به صورت گرافیکی ببینه:



بعد از انتخاب:

	۰۸:۰۰	۰۹:۰۰	۱۰:۰۰	۱۱:۰۰	۱۲:۰۰	۱۳:۰۰	۱۴:۰۰	۱۵:۰۰	۱۶:۰۰	۱۷:۰۰	۱۸:۰۰
شنبه			طراحی کامپیوتری سیستم های دیجیتال مهندسی ۲۰۳				مبانی شبکه های بی سیم مهندسی ۲۰۳				
یکشنبه							انتقال داده مهندسی		مدارهای الکتریکی مهندسی ۲۰۲		
دوشنبه							مبانی شبکه های بی سیم مهندسی ۲۰۳				
سه شنبه			انتقال داده مهندسی				مدارهای الکتریکی مهندسی ۲۰۲		تاریخ تحلیلی صدر اسلام (تاریخ اسلام) گروه معارف		
چهارشنبه			طراحی کامپیوتری سیستم های دیجیتال مهندسی ۲۰۳								

بعد از انجام اینکار اطلاعات دروس توی مرورگر ذخیره میشن و دفعات بعدی میتونین بدون مراجعه به صفحه دروستون، با انتخاب همین گزینه توی هر صفحه ای برنامه درسیتون رو ببینین.

نویسنده : جعفر آخوندعلی

رادیوهای وایرلس



از زمان های قدیم برای برقراری ارتباط با یکدیگر از راه دور نیاز به وسیله ای بود که بتواند این ارتباط را ایجاد کند. برای تحقق یافتن این امر دکل های مخابراتی را ایجاد کردند و توانستند بین مراکز مورد نیاز خود ارتباط برقرار کنند. مانند بیسیم های واکی تاکی که از دو یا چند نقطه مختلف توانستند باهم تماس داشته باشند.
حال با گذشت زمان شرکت های بسیاری در این عرصه به تولید و ساخت وسایل بیسیم و وایرلس پرداختند و توانستند ارتباط را در مقیاس وسیع و گسترده تر گسترش دهند. با توجه به پیدایش این راه نیاز بود تا هر ساختمان و یا مرکز ارتباطی قادر باشند بدون ایجاد اخلال در ارسال و دریافت اطلاعات از دیگر مراکز بر روی بنای خود دکل های را با ارتفاع بالا نصب کنند. بطور مثال شرکت های مثل آب و فاضلاب ، پلیس راه داری و جاده ، شرکت های مخابراتی و شرکت های دیگر از این دکل ها بر روی پشت بام خود نصب کرده اند تا ارتباط با دیگر مراکز مرتبط با خود را برقرار نمایند.
از این رو شرکت های که در زمینه ساخت و فروش رادیوهای بیسیم فعالیت دارند به میدان آمدند تا نیاز شرکت ها را رفع نمایند. برند های بسیاری در این امر توانستند برای خود نام و شهرتی بدست آورند. در این مقاله به بررسی یکی از برند های فعال در این زمینه پرداخته می شود تا با بیشتر با انواع این دستگاه ها آشنا شوید.
یکی از برندهای که در فروش و ارائه خدمات وایرلس در ایران فعالیت می کنند و ما قرار هست در مورد آن بحث کنیم دیوایس و دستگاه های رادیویی شرکت لیگو ویو می باشد.



لیگو ویو یک شرکتی هست که در سال ۲۰۰۷ به عنوان یک شرکت ارائه خدمات وایرلس شروع به فعالیت کرده است و توانست در کشورهای مختلف از جمله ایران بازار خود را پیدا کند. لیگو ویو با ساخت دیوایس های point to point و point to multipoint توانست نیاز شرکت های بسیاری را رفع کند.
رادیوهای لیگو ویو به چند سری تقسیم بندی شده اند که در ادامه با ۳ سری از آن را ملاحظه می نمایید.

LigoDLB



این سری از رادیوها برای کاربردهای Point to Point و Point to Multipoint در باندهای ۵ گیگاهرتز و ۲ گیگاهرتز طراحی شده اند. این سری برای سمت سرور و هم سمت کلاینت مدل های متنوعی را در اختیار کاربر های قرار است. شرکت های که اینترنت فوق العاده به کاربران می دهند مناسب است. اپراتورها می توانند در باند های مجاز شبکه های خود را به یکدیگر متصل کنند. با داشت پروتکل اختصاصی قادر هستند ارتباط خوب را در محیط های پر نویز داشته باشند. دارای پهنای باند بالا در مقیاس ۱۷۰ مگابایت در ثانیه هستند.
همچنین با بررسی امکانات و مقایسه با دیگر برندهای موجود در بازار قادر خواهند بود که بهترین انتخاب را داشته باشند.

LigoPTP

یکی از پرکاربرد ترین سری رادیوهای لیگو ویو شناخته می شود



و از لحاظ کارایی متمایز طراحی شده است. این سری از رادیوهای وایرلس ۵گیگاهرتز هستند و عملکرد بسیار بالایی دارند.مدیریت و نصب این سری از رادیو های لیگو ویو ساده می باشد. به طور اختصاصی برای پروژه های PTP مورد استفاده قرار می گیرد.

LigoPTMP



از سری محصولات جدید لیگو ویو است که از تمام جزیئات اصلی LigoPTP در این سری استفاده کرده اند. برای پروژه های یک نقطه به چند نقطه اختصاص داده شده است..در سری LigoPTMP از CPU قدرتمندی که در محصولات LigoBase استفاده می شود به کار رفته است. سرعت انتقاال در این رادیو . مگابایت در ثانیه تخمین زده شده است.
در شماره های آینده بیشتر در این خصوص مطلب قرار داده می شود تا مخاطبان علاقه مند بتوانند بیشتر با امکانات این رادیو آشنا شوند.

نویسنده : بردیا

آشغال گردی [نفوذ کثیف]

مهندسی اجتماعی همواره کارسازترین روشی است که هکر های حرفه ازش استفاده میکنند. اگه بخوام خلاصه بگم در مهندسی اجتماعی ، یک هکر از تکنیک های روانشناسی استفاده میکنه برای اینکه از شما سو ء استفاده کنه. مثلا ممکنه در حمل کردن بار از بازار تا خانه به شما کمک کند فقط برای اینکه آدرس خانه تان را پیدا کند ! یا با استفاده از حس ترحم شما را وادار به انجام کاری کند که شما در حالت معمول انجام نمیدهید. هکر ها با مهندسی اجتماعی درعوض اینکه وقت صرف شکستن پسوردتان بکنن ، کاری میکنن خودتون پسورد رو بهشون بدین! (مثل فیشینگ)



اما یکی از جالب ترین روش های مهندسی اجتماعی آشغالگردی است. در این روش هکر با جست و جو در زباله هایتان اطلاعات زیادی را بدست می آورند. خب حالا زباله هایی که من دور میریزم به چه درد میخوره و یک هکر چطوری میتونه با آشغالگردی به من نفوذ کنه ؟ برای نفوذ و هک کردن یک شخص باید در مورد اون شخص اطلاعات بدست آورد و زباله هایی که هر کسی دور میریزه میتونه اطلاعات زیادی رو راجع بهش بده.

اوکی ! بزارین برم یه سر به سطل زباله خودم بندازم ... خب رسید های بانکی که دارم دور میریزم میگه من وضعیت ثابتی تو حساب بانکی ام ندارم ولی در کل معمولا زیاد پول تو حسابم نمیمونه :) تعداد زیاد پاکت سیگار ! این یعنی به سلامتی خیلی اهمیت نمیدم پس اهل ورزش هم نباید باشم و احتمالا تا چند سال دیگه سرطان میگیرم ! باقی مونده های خوراکی نشون میده من به چه خوردنی هایی علاقه دارم و وجود رسید های بانکی زیاد از یک رستوران میتونه نشون دهنده زندگی مجردی و نداشتن علاقه یا وقت برای آشپزی باشه. دیدین چه جور اطلاعاتی دارم از خودم منتشر میکنم ؟ خیلی اطلاعات دیگه میشه از زباله های یک شخص بدست آورد و برای نزدیک شدن یا نفوذ به اون شخص استفاده کرد

نکته باحال اینه که از لحاظ قانونی آشغال گردی هیچ مشکلی نداره ! چرا ؟ چون به لحاظ حقوقی زباله هایی که هر شخص دور میریزد دیگر متعلق به او نیست و نمیتواند ادعای مالکیت بر زباله های خودش داشته باشد.

خیلی خب ، قبل تر گفتم که هکر برای نفوذ کردن به شخص نیاز داره اطلاعاتی ازش بدست بیاره ، حالا هکر با این اطلاعاتی که از طریق آشغالگردی بدست آورده چیکار میتونه بکنه ؟ بسته به نوع اطلاعات میتونه انواع سوء استفاده رو بکنه ! در بهترین حالت میتونه از این اطلاعات برای بدست آوردن رمز های عبورتان استفاده کنه .

در حال حاضر رمز عبور هایی که اکثر مردم استفاده میکنند عباراتی هستند که معمولا به آن علاقه نشان میدهند ، یا سال تولد و یا ترکیبی از اینها. آشغالگردی علایق و تمایلات هر شخص رو مشخص میکنه و یک هکر با بدست آوردن اینها و انتقال به نرم افزار هایی که برای شکستن رمز عبور استفاده میشه به اهدافشون برسند.
البته حالت های بدتری هم هست که چون خانواده نشسته از اطلاع رسانی معذوریم :)

نویسنده : حسین شفیعیان

پرسش و پاسخ برنامه نویسی

سلام.

یه سری نکات تو مطلب برنامه نویسی و استخدام مطرح شد به نظرم اومد که تو یه شماره چند تا سوال برنامه نویسی خوب هم داشته باشیم. خب بریم سراغشون!

۱- فرق abstraction با encapsulation ?

خیلی ساده با یه مثال روشن میشه.

در کپسولیشن شما عملا دسترسی به بافت قضیه ندارید ولی در ابسترکشن دارید مثل تلویزیون و ریموت کنترل.

ریموت کنترل یه ابسترک ساده میتوان در نظر گرفت که کاربر بدون دانش از این که داخل آن چه خبر است استفاده میکند(ولی میتواند دسترسی بگیرد)

اما تلویزیون هیچ ایده ای از این که فیلم و آهنگ و ... پخش میشود نداریم(حداقل فرض کنیم دسترسی محلی نداریم) و دسترسی هم نداریم و صرفا از آن استفاده میکنیم.

۲- فرق interface با abstract?

از نظر مهندسی نرم افزار هیچ فرقی این دو مفهوم ندارند. به طور کلی یک ایده دارند ولی از نظر برنامه نویسی یه سری تفاوت دارند مثلا در متغیر و داستان وراثت تو اینترفیس و پیاده سازی توابع و ....

۳- برنامه نویس سمت سرور هستید و پاسخ هایی همیشه ثابت هستند چه کار میکنید؟

از دیتابیس های جدید مثل ردیس استفاده میکنیم. راه حل ساده تر (در صورت کم بودن متغیر ها) تعریف آن در چند متغیر و استفاده با رم (همان مفهوم ردیس است)

هدف از این کار چیست ؟ قطعا دیتا در هارد کندتر از رم است. سرعت انتقال دیتا رتبه نخست برا رجیستر ها و کش CPU و بعد رم و در انتها هارد است. پس دیتابیس های عادی که با هارد سر و کار دارند به مراتب کند تر هستند.

حال چرا اطلاعات در رم نیست ؟ چون حجم اطلاعات خیلی زیاد است و هزینه ساخت و داشتن رم خیلی بیشتر از هارد است.

ممکن است فکر کنید که اگر سیستم خاموش شود چه پیش می آید یا اطلاعات رم میپرد؟ پاسخ منفی است و ردیس یک بکاپ میگیرد و مثل دیتابیس های معمولی در هارد ذخیره میکند و نگرانی ندارد !
۴- برنامه نویس سمت سرور هستید چه راهکارهایی برای ریکوئست های الکی میدهید ؟

پاسخ این سوال کمی پیچیده است اما به طور کلی استفاده از CAPTCHA ، توکن ها و authentication ها در راستای همین مطلب است. این که چجور جلوی DDOS گرفته شود شاید در پیچیده ترین حالت با هوش مصنوعی و یادگیری ماشین بهبود در عملکرد سیستم داشته باشیم.

۵- اطلاعات محرمانه ای به روی کلاینت داریم چگونه جلوی دسترسی آن را میگیرید؟

مهندسی معکوس و جلوگیری آن باز هم مبحث گسترده ای است ولی ساده ترین ایده ای که پاسخ این مسئله را میدهد رمزنگاری و سالت است.

یعنی تقریبا کاری که بعضی از اپلیکیشن های موبایلی یا تحت وب انجام میدهند. شاید تضمینی هم در کار نباشد!

۶ -OOPرا تعریف کنید و مزایا و معایبش را بگید؟
برنامه نویسی شی گرا برای راحتی برنامه نویس و مهندسین نرم افزار است مزایای فراوان که دارد و هر کدوم کاربرد خودش ولی در حالت کلی فانکشنال نوشتن به مراتب زمان بیشتری میبرد. معایب آن در عملکرد است. مثلا اگر محدویت زیادی در رم و cpu و … داشته باشیم گزینه ی مناسب قطعا OOP نیست.

تعریف خیلی ساده هم که میتوان گفت وقتی یه سری رفتار ها و داده های یکسان و قابل تعریف در یه موضوع داریم میتوان سراغ OOP برویم مثل کلاس حیوان و مثلا وراثت کلاس انسان از حیوان.(به فرض یک بازی open world داریم)
البته OOP و معماری ها و چارچوب خیلی مفصل هستند ولی در همین حد هم در حداقل مصاحبه ها به نظرم کافی باشه.
معمولا در راستای OOP باید دانش کافی از وراثت و کپسولیشن ، چند ریختی و … هم باید داشته باشیم.

۷- کلاس های یک بازی شطرنج یا بولینگ یا .. را طراحی و پیاده سازی کنید (معمولا ۴۰ تا ۶۰ دقیقه زمان)
فک میکنم سوالای معروف گوگل در استخدام باشه که یه پیاده سازی خیلی ساده با پرینت که منطق بازی کار کنه و کلاس بندی ها و معماری خوبه داشته باشه.
تو کلاس ها و معماری باید یه سری موارد را رعایت کنید مثل coupling cohesion یا single responsibility و …

۸- پیچیدگی الگوریتم … چه قدر است ؟
حتما باید مبحث پیچیدگی را مسلط باشید و بدانید الگوریتم های مورد استفاده شما چگونه عملکردی دارند. ممکن است سوالات ساده ای پرسیده شوند مثل ادغام دو آرایه نا مرتب در یک آرایه مرتب یا پیاده سازی یک الگوریتم مرتب سازی ساده، صف ، استک، لینک لیست، گراف و ….
که در نهایت شما باید بدانید چرا الگوریتم را پیشنهاد دادید و پیچیدگی چگونه است.

اگر شما هیچ اطلاعی درباره پیچیدگی الگوریتم ندارید معمولا بدترین حالت را با O نمایش میدهند و میگن و از log n تا پیچیدگی های نمایی می تواند راه حل مسائل باشد.
مثلا اگر شما یک آرایه ۱۰۰ تایی دارید و میخواهید آن را پرینت کنید.۱۰۰ بار عمل خواندن + ۱۰۰ بار پرینت دارید. یعنی در اوردر n است.(پیچیدگی زمانی)
پیچیدگی مکانی هم داریم که باید محاسبه ذخیره متغییر ها را حساب کرد.(که مثلا رم چه قدر نیاز است)

احتمالا متوجه هستید که مثلا الآن با قضیه ی سیستم های distribute مشکلات cpu را حل کرده اند ولی کمبود رم و پیچیدگی نمایی رم را کاری نمیشود کرد.

مثلا الگوریتمی در بدترین حالت ۱ سال طول میکشه خب صبر میکنیم هسته و سیستم زیاد میکنیم و به فرض به ۲ ماه کاهش میدهیم ولی وقتی رم خیلی بالایی بخواهد کار به مراتب سخت میشود یا حتی نشدنی.

این مدل سوالات رو معمولا تو مصاحبه های ایرانی میبینید. یه چیز جالبی که هست وقتی یه تکنولوژی جدیدی میاد و معروف میشه سریع همه از اون سوال میکنن و یه خورده کار مارو هم سخت میکنه. مثلا یه مدتی برا برنامه نویسی سمت سرور از mongo db سوال میکردن در صورتی که نیازی نداشتن! یا مثلا اخیرا چند جا سوال میکردن آقا مدل اسنپ را چند روزه میزنی و چجوری ..(!)

نمونه سوالات در گیت هاب و سایت های مختلف موجوده و زیاد هستن چه برا تمرین برنامه نویسی و الگوریتم و OOP چه مصاحبه و چیزای دیگه

موفق باشید

نویسنده : امین سامانی

الگوریتم‌های روتینگ (قسمت دوم)

در قسمت قبل در مورد دسته‌بندی‌های اولیه و مفاهیم ابتدایی صحبت کردیم ، حالا می‌خواهیم یک مرحله جلو بریم و در این قسمت می‌خواهیم دسته‌بندی Distance Vector و الگوریتم Rip را شرح دهیم .

معرفی پروتکل‌های روتینگ Distance Vector

همان‌طور که از نام این پروتکل‌ها پیداست ، این دسته از پروتکل‌ها از دو بخش Distance (مسافت) و Vector (جهت) تشکیل‌شده‌اند و به این وسیله مسیر خود را پیدا می‌کنند . این پروتکل‌ها به صورتی کار می‌کنند که روتر ها توپولوژی شبکه و تغییرات را به همسایه‌های خود اطلاع می‌دهند . این اطلاع‌رسانی به‌صورت Broadcast و با الگوریتم Bellman-Ford انجام می‌شود . به این صورت که روتر ها رو تمام interface های خود به‌صورت Broadcast و با آدرس ای پی به شکل ۲۵۵.۲۵۵.۲۵۵.۲۵۵ برای تمام همسایه‌های خود درخواست می‌فرستند تا از آخرین تغییرات و توپولوژی مطلع شوند و مهم‌تر از همه از جدول‌های تشکیل‌شده برای مسیریابی اطلاع پیدا کنند . پس‌ازاین که اطلاعات و تغییرات را دریافت کردند بلافاصله این تغییرات را برای همسایه‌های خود می‌فرستند . جدول‌های مسیریابی همسایه‌های خود را به‌روزرسانی می‌کنند به این صورت که پس از دریافت تغییرات Distance value روت موجود را افزایش می‌دهد و Distance value خودش را به آن اضافه می‌کند و در اختیار روتر بعدی قرار می‌دهد . نکته‌ی جالبی که هست این است که این روتر ها به این توجه نمی‌کنند که چه کسی به پیام آن‌ها گوش می‌دهد آن‌ها این پیام هارا برای همه می‌فرستند و نکته‌ی دیگر این است که حتی اگر تغییراتی نیز ایجاد نشده باشد این روتر ها بازهم هر از چند گاهی این پیام‌ها را برای بقیه نیز ارسال می‌کنند و بقیه‌ی روتر ها را از وضعیت خودآگاه می‌کنند . پروتکل‌های Distance Vector جزو ساده‌ترین پروتکل‌های مسیریابی هستند و طراحی و پیاده‌سازی آن‌ها بسیار ساده است و به منابع کمی نیاز دارند . نحوه عملکرد آن‌ها بسیار ساده است آن‌ها پیام‌های بروز رسانی را دریافت می‌کنند و مقدار metric را افزایش می‌دهند و با مقایسه‌ی مقادیر بروز رسانی در صورت نیاز جدول خود را بروز رسانی می‌کنند و پیام بروز رسانی را برای بقیه می‌فرستند . ازجمله پروتکل‌هایی که در این دسته قرار می‌گیرند می‌توان به RIP و IGRP و EIGRP که EIGRP تغریبا به‌صورت Hybrid است اشاره‌کنیم .

معرفی پروتکل مسیریابی (RIP (Routing Information Protocol

پروتکل RIP جزو اولین پروتکل‌های مسیریابی است و عملکرد بسیار ساده‌ای دارد و بر اساس معیاری به نام Hop Count رفتار می‌کند به این صورت که هر روتر را به‌عنوان یک Hop در نظر می‌گیرد و برای جلوگیری از ایجاد loop در شبکه تعداد Hop هایی که می‌تواند بگذراند را محدود می‌کند که یکی از روش‌های جلوگیری از ایجاد loop است و این تعداد ۱۵ عدد است و درصورتی‌که تعداد Hop ها یا روتر ها از ۱۵ بیشتر شود این پروتکل توانایی پیاده‌سازی در این محیط را ندارد و تعدادهای بالاتر را حالت بی‌نهایت فرض می‌کند و بسته‌ها را Drop می‌کند .

روش‌های جلوگیری از loop

۱-Split Horizon :. یعنی روتر حق ندارد مسیری که از یک روتر دیگر گرفته برای همان روتر بفرستد
۲ -Route poisoning : اگر مسیری قطع شود روتر موظف است metric را برابر۱۶ قرار دهد که همان بی‌نهایت است .
۳-Triggered Update : بدین معنی که روتر نباید صبر کند تا زمان به‌روزرسانی اصلی برسد (هر ۳۰ ثانیه یک‌بار) و هر موقع نیاز به به‌روزرسانی و ارسال پیام‌های به‌روزرسانی بود همانند Route poisoning این عمل را انجام دهد .
۴ -Hold down Timer : درصورتی‌که روتر پیام به‌روزرسانی را دریافت کرد۱۸۰ ثانیه صبر کند و سپس جدول خود را به‌روزرسانی کند .
۵ -Split horizon with Poison reverse : به این معنی که روتر حق ندارد مسیری که از یک زوتر گرفته است را دوباره به آن روتر اعلام کند مگر این‌که آن مسیر مسدود شده باشد یاRoute poison . باشد.

RIP با پروتکلUDP و روی پورت۵۲۰ کار می‌کند و معمولاً در شبکه‌های کوچک مثلLAN استفاده می‌شود و به‌عنوان یک مسیریاب داخلی یاInterior Routing نامبرده می‌شود . به مرور زمان ایراد هایRIP گرفته شد و سبب بروز تغییراتی شد و به همین دلیل ورژن های مختلفی از آن به وجود آمد که در ادامه به تفاوت ها و تغییرات این نسخه ها اشاره میکنیم .

پروتکل‌های مسیریابی Calssful vs Classless

Classful : پروتکلهایی که هیچ گونه اطلاعاتی در مورد Subnet Mask . منتقل نمی‌کنند
Classless : این پروتکلها برخلاف مورد قبل هنگام تبادل اطلاعات Subnet Mask . را نیز منتقل می‌کند

نسخه‌های مختلف RIP

انواع نسخه‌های RIP :
- Version ۱
- Version ۲
- RIPng (RIP next generation)

ویژگی‌های RIP version ۱ :
- یک پروتکل Classful است و از VLSM پشتیبانی نمی‌کند
- امکان Authentication نیست
- به‌روزرسانی‌ها را به‌صورت Broadcast ارسال می‌کند

ویژگی های RIP version ۲ :
- یک پروتکل Classless است و از VLSM پشتیبانی می‌کند
- امکان Authentication دارد
- به‌روزرسانی‌ها را به‌جای Broadcast به‌صورت multicast به آدرس ۲۲۴.۰.۰.۹ می‌فرستد.

ویژگی‌های RIPng :

- پشتیبانی از IPv۶
- از پروتکل UDP با شماره پورت ۵۲۱ استفاده می‌کند.

RIPv1	RIPv2
Classful routing protocol	Classless routing protocol
Uses broadcast to send periodic updates	Uses multicast to send periodic updates
No support for authentication	Supports authentication
No support for VLSM/CIDR	Support for VLSM/CIDR

نحوه پیاده‌سازی

برای پیاده‌سازی این پروتکل در ابتدا لازم است که به روتر ها بفهمانیم از چه پروتکلی استفاده می‌کنیم برای این کار از دستور زیر استفاده می‌کنیم :

Router (config)#router rip

سپس باید شبکه‌هایی که اطراف این روتر قرار دارند را برایش مشخص کنیم که برای این کار همانند زیر عمل می‌کنیم :

Router (config-router)#network ۱۰.۱۰.۱۰.۱۰

با استفاده از دستورهای دیگر می‌توان میزان زمان و فاصله و مسیرها را تنظیم کرد .

Router (config)#router rip
’Router (config-router)#maximum-paths ‘one number between ۱ to ۶
Router (config-router)#traffic-share min across-interfaces

که به این وسیله می‌توان حداکثر تعداد مسیر هارا تعیین کرد .

’Router (config-router)#timers basic ‘update’ ‘invalid’ ‘Hold on’ ‘Flush

این دستور برای تنظیم زمان‌ها در RIP است که زمان اول و دوم و سوم در موردش صحبت کردیم اما عدد چهارم هم Flush timer است که مربوط به کانکشن هایی است که قطع‌شده و از جدول پاک می‌شود .

در این قسمت در مورد روتینگ RIP صحبت کردیم در قسمت‌های بعدی در مورد روتینگ های پیشرفته تر صحبت میکنیم .

نویسنده : سیروس فتح اللهی دهکردی – کارشناس مهندسی فناوری اطلاعات – فعال در حوزه ی شبکه های کامپیوتری و علوم مهندسی کامپیوتر – علاقه‌مند به زمینه‌های امنیت و مدیریت اطلاعات و فناوری های نوین

ز گهواره تا گور دانش بجوی

